Elisabeth Fughe
Matrikelnummer: 5263769
s3499227@stud.uni-frankfurt.de

Bachelorarbeit (B.Sc. - Informatik)

tbd

Elisabeth Fughe

Abgabedatum: tbd 2019

FIAS - Frankfurt Institute for Advanced Studies
Prof. Dr. Nils Bertschinger

Erklärung

gemäß Bachelor-Ordnung Informatik 2011 §25 Abs. 11

Hiermit erkläre ich Frau

_____

Die vorliegende Arbeit habe ich selbständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst.

Frankfurt am Main, den _____

_____

Unterschrift

Zusammenfassung

Abschließend wird in Kapitel 1 …

Inhaltsverzeichnis

1 Einleitung

tbd

2 Verwendete Methoden

tbd

2.1 Bayessche Modellierung

Die Bayessche Statistik untersucht mittels bayesscher Wahrscheinlichkeiten und dem Satz von Bayes Fragestellungen der Stochastik. Anders als in der klassischen Statistik, die unendlich oft wiederholbare Zufallsexperimente voraussetzt, steht die Verwendung und Modellierung von Wahrscheinlichkeitsverteilungen im Vordergrund.

Es gilt, das beobachtete Daten

$$x = (x_1, ..., x_n)$$

mittels bedingter Wahrscheinlichkeiten in Beziehung zu unbekannten Parametern

$$\theta = (\theta_1, ..., \theta_m)$$

stehen. Sodass die gemeinsame Wahrscheinlichkeitsdichte

$$p(x, \theta) = p(x|\theta) \cdot p(\theta)$$

durch die a-prori-Verteilung unbekannter Parameter $p(\theta)$ und den Erkenntnissen aus dem Datensatz $p(x|\theta)$ berechnet werden kann. Durch den Satz von Bayes kann dann die a-posteriori-Verteilung unbekannter Parameter

$$p(\theta|x) = \frac{p(x|\theta) \cdot p(\theta)}{p(x)}$$

ermittelt werden [1].

Die a-posteriori-Verteilung enthält somit Informationen über die unbekannten Parameter durch die Kombination der a-prior Verteilung mit den Informationen, die aus den beobachteten Daten gewonnen wurden.Sie wird zur Punktschätzung und Schätzung von Konfidenzintervallen genutzt.

So sind bayessche Modelle, im Gegensatz zur klassischen Statistik, auf kleineren Datensätzen anwendbar, dort ergibt sich jedoch eine breite Wahrscheinlichkeitsverteilung, die somit unter Umständen eine geringe Genauigkeit aufweist.

2.2 Markov Chain Monte Carlo (MCMC)

In der bayesschen Statistik beschreibt die a-posteriori-Verteilung die Unsicherheit der unbekannten Parameter, die anhand beobachteter Daten geschätzt wurden. Mit der Markov Chain Monte Carlo (MCMC) Methode kann die a-posteriori-Verteilung und somit die unbekannten

Parameter untersucht werden [3]. Dazu wird eine Markov-Kette entworfen, deren langfristiges Gleichgewicht der Wahrscheinlichkeitsverteilung(Ziel Wahrscheinlichkeitsdichte - target density - a-posteriori-Verteilung) entspricht. Anschließend wird diese Markov-Kette solang simuliert bis sie mit einer entsprechenden Sicherheit, das Gleichgewicht erreicht hat. Dann wird der finale Zustand der Markov-Kette als Teil der Zufallsstichprobe/des Samples notiert [4]. Die Markov-Kette generiert so eine Reihe von Modell-Realisierungen, die zufällig aus der a-posteriori-Verteilung gezogen werden [3].

Ein weit verbreitetes MCMC-Verfahren ist der Metropolis-Hastings-Algorithmus. Der Algorithmus startet an einem zufälligen Punkt im zu untersuchenden Vektorraum/ in der zu untersuchenden Verteilung (a-posteriori-Verteilung). Dann wird eine Schrittweite zufällig mit Hilfe einer symmetrischen Wahrscheinlichkeitsverteilung gewählt. Der Schritt wird abgelehnt oder akzeptiert auf Grundlage der Wahrscheinlichkeit der neuen Position im Verhältnis zur alten Position [3]. So wird sicher gestellt, dass die Markov-Kette, von jedem Punkt aus gegen das langfristige Gleichgewicht, der Ziel-Wahrscheinlichkeitsdichte, konvergiert [1]. Der Metropolis-Hastings-Algorithmus ist sehr einfach zu implementieren und liefert gute Ergebnisse insbesondere bei stark korrelierten Parametern [3].

2.3 Hamiltonian Monte Carlo Sampling (HMC)

tbd

2.4 Stan und R

Stan ist eine Open-Source Plattform für statistische Modellierung und high-performance Berechnungen. Stan ist für alle in der Datenanalyse weit verbreiteten Sprachen (R, Python, shell MATLAB, Julia, Stata) verfügbar und läuft auf den gängigen Betriebssystem (Linux, Mac, Windows) [2]. Jedes Stan Programm startet mit dem Data-Block, der definiert welche Daten-Inputs benötigt werden um das Modell zu fitten . Die Variablen in diesem Block entsprechen also den tatsächlichen Beobachtungen. Daran anschließend kommt der Parameter-Block, der die unbekannten Parameter definiert. Zum Schluss kommt der Modell-Block, der Berechnungsvorschrift der Wahrscheinlichkeitsdichte des Modells [2]. Der Stan-Code der Modelle, die in dieser Arbeit genutzt wurden befindet sich im Anhang und wird in Kapitel 3 näher erläutert.

Die Modelle wurden mittels *rstan* in R gefittet. *Rstan* ermöglicht es Stan-Modelle in R zu kompilieren, zu testen, und zu analysieren. Außerdem wurden die Pakete *tidybayes* und *tidyverse* genutzt, um die Plots zu generieren. Die tidy* Pakete zeichnen sich dadurch aus, dass sie der komplexen Datenstruktur des berechneten Fits, Informationen leicht entnehmen kann und in einer Datenstruktur zu Verfügung stellt, die anschließend z.B. mittels *ggplot* leicht visualisiert werden kann.

3 Die Modelle

tbd

3.1 GARCH

tbd

3.2 Vikram & Sinha (VS)

tbd maybe

3.3 Franke & Westerhoff (FW)

tbd

3.4 AL herd walk (AL)

tbd

4 Simulationen

tbd

4.1 Daten & Vorhersagen

S&P 500 data in USD from finance.yahhoo:
-daily prices - calculated into return and finally into log return as models expect log return as inputs
-exporting data with dates like Jan 1 2000 to Jan 1 2008 yahoo automatically uses last working day at stock exchange
- always used 30 Predictions - inaccurate as days per month change
=> this all leads to inaccurancy which is ok as the goal is to see the tendency in which way predictions are shifting

monthly prediction during a year in which a major crisis happend: bank crisis 2008 & dotcom 2000?
used 8 years to predict the year 2008 on a monthly basis: e.g. data from Jan 1 2000 to Jan 1 2008 used to predict Jan 2008

4.2 Ergebnisse

tbd

Literatur

[1] N. Bertschinger, I. Mozzhorin und S. Sinha. „Reality-check for Econophysics: Likelihood-based fitting of physics-inspired market models to empirical data“. In: *CoRR* abs/1803.03861 (März 2018). arXiv: 1803.03861. URL: http://arxiv.org/abs/1803.03861.

[2] B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li und A. Riddell. „Stan : A Probabilistic Programming Language“. In: *Journal of Statistical Software* 76.1 (Jan. 2017). DOI: 10.18637/jss.v076.i01.

[3] K. M. Hanson. „Markov chain Monte Carlo posterior sampling with the Hamiltonian method“. In: Bd. 4322. 2001, S. 4322 - 4322 –12. DOI: 10.1117/12.431119. URL: https://doi.org/10.1117/12.431119.

[4] W.S. Kendall, F. Liang und J.S. Wang. *Markov chain Monte Carlo: Innovations and Applications*. Bd. 7. World Scientific Publishing Co Pte Ltd, 2005.